



# BOTALK

보험계약시 불완전판매 실시간 판단서비스

조미나	alsk7091@gmail.com
김기원	orizinekgw@gmail.com
최수진	dalgon02121@naver.com

# 1. 'BOTALK' 서비스의 개요

## 1-1. 서비스의 개요

- \* 은퇴 설계가 필요한 **중·장년층을 대상**으로 한 지식서비스
- \* 보험 상품의 불완전판매 여부 판단, 누락된 키워드 알림, 약관의 시각화를 제공하는 어플리케이션 형태의 서비스

불완전판매 방지 서비스

BOTALK



## 2. 불완전판매 문제 인식

### 2-1. 문제 발생 배경

- \* 최근 DLF(파생결합펀드)와 라임 사태 등 금융상품의 불완전판매 문제가 불거지고 있음
- \* 특히 보험상품의 소비자들은 보험설계사의 설명에 전적으로 의존하기 때문에 불완전판매 가능성이 높음



HOME > 경제 > 금융

#### 생명보험 민원 지난해 3만 건, 불완전판매 여전

임해원 기자 | 승인 2020.02.21 10:56 | 댓글 0



< 생명보험 민원 발생 최상, 최하 비교 >

구 분	최상위	업계평균	최하위	비고
민원 건수	삼성생명 7,612건	1,245건	저브라이프 29건	
환산 건수	KDB생명 93.00건	34.34건	하나생명 7.73건	
판매(영업) 민원	DB생명 92%	57.9%	교보생명 28.3%	
보험금(지급) 민원	교보, 한화 55.2%	34.4%	MET생명 0.6%	
변액보험 민원	MET생명 67.8%	18.2%	농협 0.0%	
종신보험 민원	KB생명 79.7%	39.5%	라이나 1.4%	

\* 2019년 '생명보험회사 자체통계기준(생명보험협회 공식실 자료 참조)

자료=금융소비자연맹

지난해 생명보험 민원이 3만 건 가까이 발생한 가운데, 삼성·한화·교보 등 국내 상위 3사가 절반 이상을 차지한 것으로 집계됐다.

금융소비자연맹(이하 '금소연')이 지난 20일 발표한 '생명보험사 2019년 민원 발생 현황 분석' 보고서에 따르면, 지난 한 해 동안 발생한 생명보험 민원은 2만8630건으로 계약 10만 건당 발생 건수는 788건이었다.

## 2. 불완전판매 문제 인식

### 2-2. 문제 제기

- \* 보험 계약 상 소비자와 판매자 간의 정보 비대칭 문제가 존재함
- \* 전문적인 용어와 난해한 설명으로 인해 일반적인 소비자, 특히 중·장년층이 약관을 이해하는데 있어 많은 어려움을 겪음



### 3. 서비스를 통한 해결방안 제시

#### 3-1. 해결방안 제시

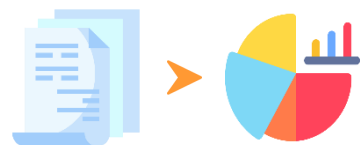
---



**불완전판매 여부 판단**  
: 소비자가 능동적으로 대응할 수 있도록 함



**누락된 중요 키워드 알림**  
: 소비자들의 정보비대칭을 해소함



**약관의 시각화**  
: 소비자의 폭넓은 이해를 도움

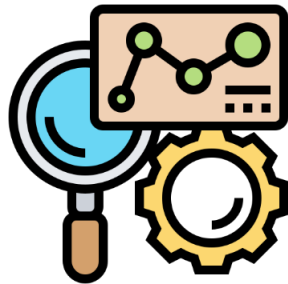
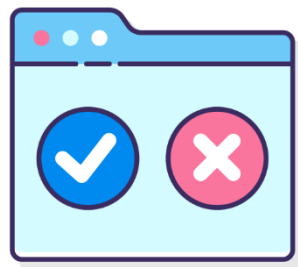
---

## 4. 서비스 구현 과정

### 4-1. 불완전판매 여부 판단 서비스

---

- \* 수집한 문장 가공
  - \* 자연어 처리를 통해 불완전판매 여부 판단
- 



## 4. 서비스 구현 과정

### 4-2. 중요 키워드 알림 서비스

- 
- \* 설계사가 보험 상품을 판매하는 데 있어 중요한 키워드 도출
  - \* 해당 키워드가 누락되었을 때, 그 키워드를 알려주는 서비스
- 

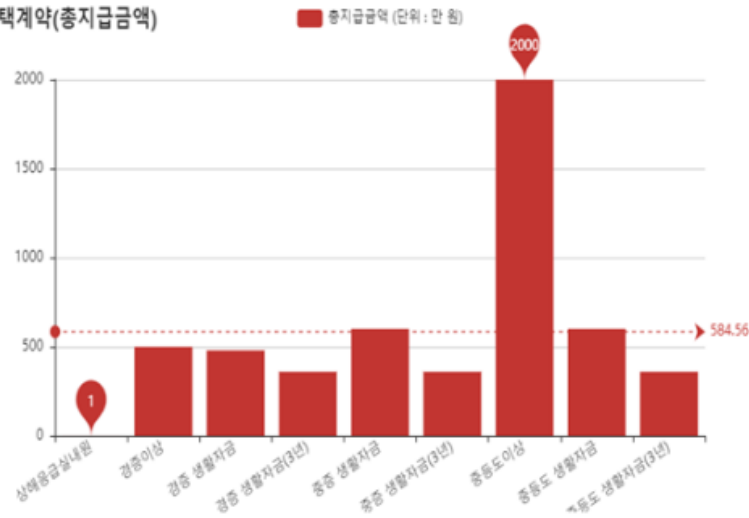


## 4. 서비스 구현 과정

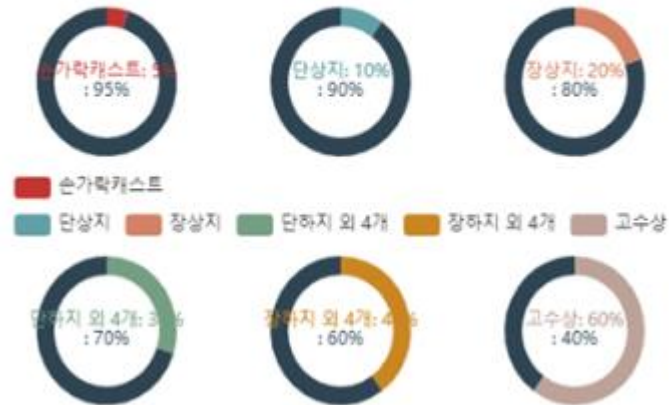
### 4-3. 약관의 시각화 서비스

- \* 생명보험협회의 설문조사에 따라 보험 가입 시 중요한 요소 도출
- \* 중요한 요소들을 한 눈에 보기 쉽게 시각화

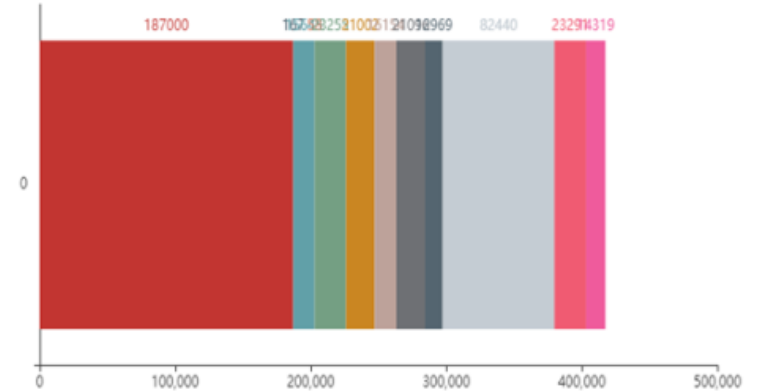
선택계약(총지급금액)



신깍스치료비 지급률



중등(기본계약) 상해유급실내원 신깍스치료비 신깍스치료비 경증이상 경증 생활자금 경증 생활자금(3년) 중등 생활자금 중등 생활자금(3년) 중등도 이상 중등도 생활자금 중등도 생활자금(3년)





## 5. 유사서비스

### 5-1. 은행권의 불완전판매 방지 AI 서비스

\* 신한은행, 하나은행, 우리은행의 불완전판매 모니터링 서비스 상용화 예정



AI 활용한 '불완전판매' 근절방안 \*자료=각 은행

신한은행	상품 가입 시 고객 반응을 녹취해 AI로 실시간 모니터링. 시스템 추진 중
하나은행	고객 필체를 AI로 인식. 상품가입서 자서 누락 방지. 상반기 상용화 예정
우리은행	신규 상품·서비스 출시 때 소비자 보호 항목 등 약관을 AI 활용해 검토

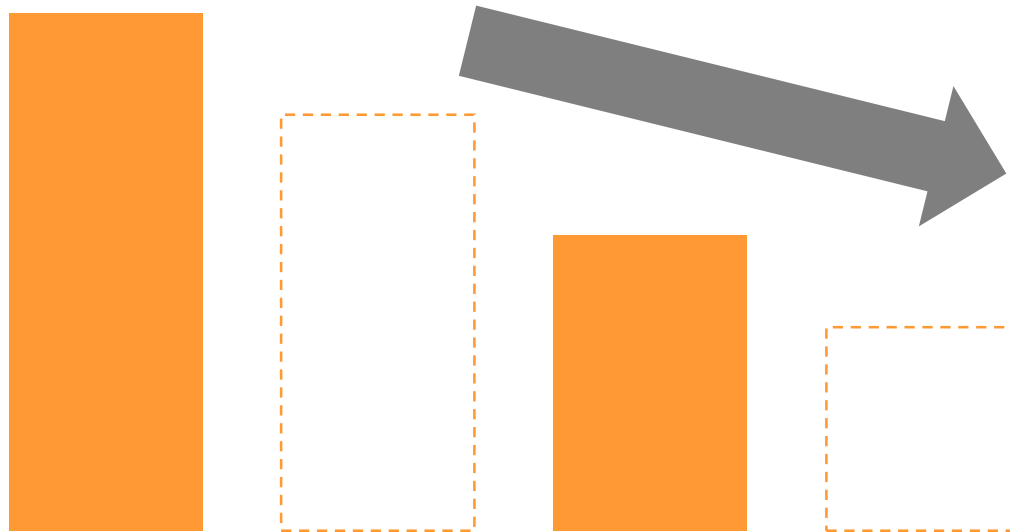
강화된 판매 절차 설명의무 \*자료=금융위원회

- 판매자와 투자자 모두 자필과 육성으로 상품 특성과 위험성 숙지
- 판매자가 투자자 성향 분류하거나 대신 기재 시 임원 해임, 6개월 이내 업무정지 등 제재
- 만 65세 이상 및 투자성향 부적합 투자자는 숙려기간 중 별도 승낙 표시해야 투자 확정

## 6. 서비스의 사회적 기여 방안

### 6-1. 불완전판매 감소

- 
- \* 고객이 스스로 설계사의 불완전판매 행위를 인지할 수 있음
  - \* 이에 따라 고객은 상품 구입 여부를 주체적으로 선택할 수 있음
  - \* 결과적으로 불완전판매가 감소함
- 



## 6. 서비스의 사회적 기여 방안

### 6-2. 소비자와 판매자 간 정보 비대칭 문제 해소

- 
- \* 고객에게 상품 약관의 내용을 한 눈에 보기 쉽게 보여줌
  - \* 따라서 보험 상품의 소비자와 판매자 간 정보 비대칭 문제 해소
- 



- \* Okt를 활용하여 수집한 데이터를 형태소 분석
- \* sklearn의 train\_test\_split을 활용하여 학습셋과 트레인셋 분리

```
def tokenizer(text):
    okt = Okt()
    return okt.morphs(text)
```

```
def step1_data_preprocessing() :
    # 수집한 데이터를 읽어오기
    df = pd.read_excel('c:/mina/insu/sentence(no_label).xlsx')

    # 리스트로 바꿔줘
    text_list = df['sentence'].tolist()
    star_list = df['score'].tolist()

    # 학습셋, 테스트셋 분리
    from sklearn.model_selection import train_test_split

    # 데이터의 80%는 학습셋이고 20%는 테스트셋
    text_train, text_test, star_train, star_test = \
        train_test_split(text_list, star_list, test_size=0.2, random_state=0)

    return text_train, text_test, star_train, star_test
```

```
step1_data_preprocessing()
```

(["예정 이율"이라는게 있어요 고객님, 보험사에서 미래에 보험금을 지급할 때를 대비해서 고객님께서 납입한 보험료를 적립해두는데, 보험료 납입하신 시점이랑 고객님의 보험금을 지급받으신 시점에 시간 차가 있는데, 그 동안 보험사에서 적립된 보험료를 운용할 수가 있으세요. 그 때 운용을 하면서 수익이 생길 수 있잖아요, 이때 수익을 미리 예상해서 일정한 비율로 보험료를 고객님들께 할인해드리고 있어요. 이때 발생하는 할인율을 '예정이율'이라고 하는데 일반적으로는 예정이율이 높으면 보험료가 저렴해지고, 낮으면 보험료가 조금 올라가세요. ",

‘재물관련 실손보상 말씀하시는 거예요 고객님, 이 상품 같은 경우는 교체비용이나 수리복구비용을 감가한 손해액에 대해서 보험가입금액이 보험가액보다 적은 일부보험이라도 비례보상을 하지 않고 보험가 입금액을 한도로 전액을 지급하는 상품이에요.’

‘확실히’.

모든

'업계최고'

‘상해 관련 보장같은 경우에 고객님께서 질병이나 체질적인 요인이 있고, 경미한 외부 요인으로 발병하거나 그 증상이 더욱 악화되었을 때에는 상해관련 보험금이 지급되지 않습니다.’

'매월 10%는 무조건 나와요'.

'따라서 고객님께서 기존에 가입중이신 상품에서 보장되지 않는 부분에 대해서도 보장  
이 됩니다.'

## 참고) 서비스 구현 모델

- \* TF-IDF를 활용하여 형태소화 된 문장을 학습시킬 수 있도록 벡터화
- \* LogisticRegression을 활용한 회귀분석
- \* 감성분석 모델 생성 및 모델의 정확도 도출

```
def step2_learning(X_train, y_train, X_test, y_test) :  
  
    tfidf = TfidfVectorizer(lowercase=False, tokenizer=tokenizer) #클래스 등장  
    # 드디어 로지스틱 등장!!  
    logistic = LogisticRegression(C=10, penalty='l2', random_state=0)  
    # 하나로 묶을 것임  
    pipe = Pipeline([('vect', tfidf), ('clf', logistic)])  
  
    # 드디어 학습한다  
    pipe.fit(X_train, y_train)  
  
    # 학습 정확도 측정  
    y_pred = pipe.predict(X_test)  
    print(accuracy_score(y_test, y_pred))  
  
    # 학습한 모델을 저장한다  
    with open('pipe.dat', 'wb') as fp: #write 쓴다  
        pickle.dump(pipe, fp) # 피클로 저장한다  
  
    print('저장 완료') # 학습된 모델 저장 완료
```

```
def learning() : # 감성분석 모델 생성  
    text_train, text_test, star_train, star_test = step1_data_preprocessing()  
    step2_learning(text_train, star_train, text_test, star_test)
```

```
learning()
```

0.7628865979381443

저장 완료



감사합니다