

Design Philosophy: Learning Through Experimentation

ICF Estimation Project · 2025-12-06

This document captures the design philosophy and approach that guides this project - a focus on learning, experimentation, and understanding rather than perfect metrics or production optimization.

0.1. The Core Insight

This is a fun experimental project, not a production system.

The goal is to **learn, experiment, and build something interesting** - not to achieve perfect metrics or production-grade performance.

0.2. What Makes a Good Experimental Project?

0.2.1. 1. Learning & Understanding (Most Important)

- Understand how the model learns from bytes
- Discover what patterns it finds
- Explain why it works (or doesn't)
- Learn about language structure

Examples of what we've learned:

- "The model learns that 'ing' endings are common"
- "Multi-loss training improves ranking but slows convergence"
- "Character patterns capture morphological structure even without explicit linguistic knowledge"

0.2.2. 2. Experimentation (Very Important)

- Try different architectures
- Try different loss functions
- Compare approaches
- See what happens

Examples of experiments:

- "What if we use hierarchical embeddings?"
- "Does contrastive loss help?"
- "How does box embedding compare to standard?"

0.2.3. 3. Interesting Results (Nice to Have)

- Model does something unexpected
- Discover interesting patterns
- Find surprising behaviors

Examples of interesting results:

- "Model predicts high ICF for portmanteaus even if not in training"
- "The model recognizes 'flimjam' as English-like despite never seeing it"
- "Multi-loss helps ranking but hurts absolute accuracy"

0.2.4. 4. Reasonable Performance (Good Enough)

- Model learns something (not just mean)
- Generalizes to unseen words
- Fast and small

Not Required:

- Perfect accuracy
- Beating benchmarks

- Production-grade performance

0.3. What We Focus On

0.3.1. Do This

1. Experiment with architectures

- Try hierarchical embeddings
- Try box embeddings
- Compare sizes vs performance

2. Experiment with training

- Try multi-loss
- Try curriculum learning
- See what works

3. Understand the model

- What patterns does it learn?
- Why does it work?
- What are its limitations?

4. Have fun

- Try weird things
- See what happens
- Learn something new

0.3.2. Don't Worry About

1. Perfect metrics

- MAE doesn't need to be < 0.1
- Spearman doesn't need to be > 0.8
- Good enough is good enough

2. Exhaustive optimization

- Don't need to try every combination
- Don't need perfect hyperparameters
- Don't need to squeeze every % of accuracy

3. Production deployment

- Not building for production
- Not optimizing for scale
- Not worrying about edge cases

4. Beating state-of-the-art

- Not competing with large models
- Not trying to be best-in-class
- Just trying to learn

0.4. Realistic Expectations

0.4.1. What's Achievable (Research-Based)

For less than 50k parameter models:

- 50-70% of baseline accuracy
- MAE 0.1-0.3 (high-freq) / 0.3-0.5 (full vocab)
- Spearman 0.6-0.8 (theoretical bound 0.18-0.19 for character-level)

Our Targets (Realistic):

- MAE < 0.25 (high-freq) / < 0.5 (full vocab) ✓

- Spearman > 0.15 (approaching theoretical bound) ✓
- Jabberwocky 3/5+ tests pass ✓
- Fast and small ✓

0.4.2. What's Not Realistic

- MAE < 0.1 (too strict for tiny models)
- Spearman > 0.8 (theoretically impossible for character-level)
- Perfect accuracy (impossible with less than 50k params)
- Production-grade robustness (not the goal)

0.5. The Experimental Manifesto

1. Learning > Perfection

- Understand how it works
- Learn from failures
- Document insights

2. Experimentation > Optimization

- Try interesting things
- See what happens
- Don't over-optimize

3. Interesting Results > Perfect Metrics

- Celebrate discoveries
- Learn from surprises
- Share what you learned

4. Fun > Production

- This is for fun
- Not for deployment
- Enjoy the process

0.6. What Success Looks Like

0.6.1. Success = Learning Something Interesting

Examples:

- “I learned how CNNs extract morphological patterns from bytes”
- “Box embeddings are cool but don't help much for this task”
- “Multi-loss training improves ranking but convergence is slower”
- “The model generalizes to portmanteaus even without training data”
- “The model recognizes ‘flimjam’ as English-like despite never seeing it”

0.6.2. Success ≠ Perfect Metrics

Not success:

- “Achieved MAE < 0.1” (too strict)
- “Beat all benchmarks” (not the goal)
- “Production-ready” (not needed)

0.7. The Bottom Line

This is a fun project. The goal is to learn, experiment, and build something interesting - not to achieve perfect metrics or production-grade performance.

Focus on:

- Understanding how it works

- Trying interesting things
- Learning from experiments
- Having fun

Don't focus on:

- Perfect accuracy
- Exhaustive optimization
- Production concerns
- Benchmark scores

Remember: The best experimental projects teach you something new and leave you with interesting insights, not perfect metrics.