I will try to explain more clearly.

I short describe my Post class and next I will give an example issue using manage.py shell.

```
class PublishedManager(models.Manager):
    def get_queryset(self):
        return super(PublishedManager,
                     self).get_queryset().filter(status='published')


class Post(models.Model):
    STATUS_CHOICES = (
        ('draft', 'Roboczy'),
        ('published', 'Opublikowany'),
    )
    title = models.CharField(max_length=250)
    slug = models.SlugField(max_length=250,
                            unique_for_date='publish')
    author = models.ForeignKey(User,
                               related_name='blog_posts',
on_delete=models.CASCADE)
    body = models.TextField()
    publish = models.DateTimeField(default=timezone.now)
    created = models.DateTimeField(auto_now_add=True)
    updated = models.DateTimeField(auto_now=True)
    status = models.CharField(max_length=10,
                              choices=STATUS_CHOICES,
                              default='draft')
    objects = models.Manager()  # Menedżer domyślny.
    published = PublishedManager()  # Menedżer niestandardowy.
    # Obsługa tagów.
    tags = TaggableManager()

    class Meta:
        ordering = ('-publish',)

    def __str__(self):
        return self.title

    def get_absolute_url(self):
        return reverse('blog:post_detail',
                       args=[self.publish.year,
                             self.publish.strftime('%m'),
                             self.publish.strftime('%d'),
                             self.slug])
```

Shell example:

```
>>> from blog.models import Post
>>> published_post = Post.published.all()
>>> published_post
<QuerySet [<Post: Składnia Markdown!!>, <Post: bbbb>, <Post: ggaga>, <Post: To
ja narcyz>, <Post: Piłka nożna>, <Post: Nowy tytuł posta>, <Post: Django 2.0>]>
>>> published_post[6].tags.all()
<QuerySet [<Tag: django>, <Tag: muzyka>, <Tag: jazz>]>
```

And now I want to filter my results by tag:

```
>>> filtered_post = published_post.filter(tags__name__in=["muzyka"])
```

In Django 1.11 i have this results:

```
>>> filtered_post
<QuerySet [<Post: Nowy tytuł posta>, <Post: Django 2.0>]>
```

So everything is ok.

But in django 2.0 i have this:

```
Traceback (most recent call last):
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/core/handlers/exception.py", line 35, in inner
response = get_response(request)
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/core/handlers/base.py", line 128, in _get_response
response = self.process_exception_by_middleware(e, request)
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/core/handlers/base.py", line 126, in _get_response
response = wrapped_callback(request, *callback_args, **callback_kwargs)
File "/home/prorok/Django/mysite/blog/views.py", line 95, in post_detail
similar_posts =
Post.published.filter(tags__in=post_tags_ids).exclude(id=post.id)
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/db/models/manager.py", line 82, in manager_method
return getattr(self.get_queryset(), name)(*args, **kwargs)
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/db/models/query.py", line 836, in filter
return self._filter_or_exclude(False, *args, **kwargs)
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/db/models/query.py", line 854, in _filter_or_exclude
clone.query.add_q(Q(*args, **kwargs))
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/db/models/sql/query.py", line 1252, in add_q
clause, _ = self._add_q(q_object, self.used_aliases)
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/db/models/sql/query.py", line 1276, in _add_q
split_subq=split_subq,
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/db/models/sql/query.py", line 1152, in build_filter
lookups, parts, reffed_expression = self.solve_lookup_type(arg)
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/db/models/sql/query.py", line 1015, in solve_lookup_type
_, field, _, lookup_parts = self.names_to_path(lookup_splitted, self.get_meta())
File "/home/prorok/Django/Django/lib/python3.6/site-
packages/django/db/models/sql/query.py", line 1390, in names_to_path
pathinfos = field.get_path_info(filtered_relation)
TypeError: get_path_info() takes 1 positional argument but 2 were given
```